

PROCEDURE 2020	DATABASE ADMINISTRATION	DATE: 09/22/81
----------------	-------------------------	----------------

TITLE: IDD Usage
------------------

Purpose
---------

To establish standard procedures for usage of the Integrated Data Dictionary (IDD)

Definitions
-------------

See Procedure #2021 - Data Base Glossary.

IDD Availability
------------------

The Integrated Data Dictionary is available for storage of IDMS definitions and COBOL copy statements (both IDMS and non-IDMS). The global dictionaries are: TESTDICT (test), QADICT (QA), and PRODDICT (production). In addition, several of the larger departments and/or projects may have dictionaries for their exclusive use.

Responsibilities
------------------

DBAS staff is responsible for controlling and monitoring IDD usage, and for dictionary backup and recovery. Production dictionaries are backed up once a week, while test and QA dictionaries are backed up three times a week.

Only DBAS staff may update QA and production dictionaries.

Development staff may add, modify, and delete entities on the test dictionaries, as indicated in the next section, "Naming Conventions". Dictionaries can be updated via two online utilities (IDD and DME) or a batch utility (IDMSDDDL). See the appropriate IDMS manuals for syntax for these utilities.

Developers have the option of putting security on their IDD entities, to limit who can update them. Several dictionary reports are available to assist developers in managing their IDD entities.

## Naming Conventions

IDD entities must be named according to the specific pattern for the entity, as shown below. Within the patterns:

### is the ITD user accounting number (eg. 123)

@ is the subsystem letter (eg. A)

&&& is any meaningful combination of letters and digits (eg 987 or 8A3)

etc is any meaningful combination of characters which conforms to COBOL identifier naming rules (eg. UPDATE-ORDERS)

rec is a record name (database, work, or COBOL copy record)

xxxx is a valid CICS TRANSACTION ID (eg. LMNO)

% is for type of access, 'u' for update, 'r' for retrieval

? is for segment type - 's' for standard, 'i' for index,  
'n' for temporary, 'v' for VDB,  
other letters for extra test files

\*\* is for CV number (eg. 20, 22, etc.)

! is for database type - 't' for test, 'q' for QA, 'p' for producti

-----

Entities that developers can create - the naming patterns are:

Entity Type	Patterns	Examples
Application	AP###@&&&	AP123A98
COBOL copy record	etc	LICENSE-RECORD
Code Table	C###@&&&	C123A987
Dialogue	D###@&&&	D123A987
Database record (ver 1)	&&&&&&	DEPT
Edit Table	E###@&&&	E123A987
Element	rec-etc	DEPT-ADDRESS-1
Function	F###@&&&	F123A987
File	X###@&&&	X123A987
Help Screen Map	H###@&&&	H123A987
MAP	M###@&&&	M123A987
Process module	D###@&&&-etc	D123A987-SHOW-ACTION
Work Record	W###@&&&	W123A987

Names of entities should match in characters 2 through 7 where possibl

-----

Entities that only DBAS staff can create - the naming patterns are:

Entity Type	Patterns	Examples
Clist	B###@&&&	B123A987
Program	S###@&&&	S123A987
Task Code	xxxx	DBPS
View	etc	IFAS_LDGR
Segment	K###@&&?	K479H01S
DBname	DBxxxx?!	DBMNSCST
Area	etc-AREA	COMPANY-AREA
Set	rec-rec	DEPT-EMPLOY
Index set	IX-rec	IX-DEPT
Schemas		
Base schema	xxxxSCHM	TXIXSCHM
View schema	xxxxVIEW	TXIXVIEW
Subschemas		
CV (on-line)	A###@%&&	A473LU01
Local Mode	L###@%&&	L473LR00
DMCL's		
CV (on-line)	DMCL**	DMCL24
Local Mode	LD###@&&	LD474M00
DBtable's		
CV (on-line)	DBT**	DBT24
Local Mode	LT###@&&	LT474M00

] Version Numbers ]
] _____ ]

The default version number for all IDD entities is '1'. If you have only one version, it should be version '1'. For database records, versions 50, 96, and 97 are reserved for DBAS use. Developers may use any other version numbers. Only version '1' of all entity types will be migrated to the QA and production dictionaries.